



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/735,855	12/16/2003	Jacob K. Gotwals	42339-192084	3162
26694	7590	07/24/2007		
VENABLE LLP P.O. BOX 34385 WASHINGTON, DC 20043-9998			EXAMINER CHEN, QING	
			ART UNIT	PAPER NUMBER
			2191	
			MAIL DATE	DELIVERY MODE
			07/24/2007	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action SummaryApplication No. ¹

10/735,855

Applicant(s)

GOTWALS ET AL.

Examiner

Qing Chen

Art Unit

2191

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 20 June 2007.
- 2a) ☒ This action is **FINAL**. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-24 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-24 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☒ The specification is objected to by the Examiner.
- 10) ☒ The drawing(s) filed on 20 June 2007 is/are: a) ☒ accepted or b) ☐ objected to by the Examiner.
 Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
 Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413) |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | Paper No(s)/Mail Date. _____ |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

1. This Office action is in response to the amendment filed on June 20, 2007.
2. **Claims 1-24** are pending.
3. **Claims 1, 8, 12, and 17-24** have been amended.
4. The objections to the drawings are withdrawn in view of Applicant's amendments to the specification.
5. The objection to the abstract is withdrawn in view of Applicant's amendments to the abstract.
6. The objections to the specification are withdrawn in view of Applicant's amendments to the specification.
7. The objections to Claims 8, 12, 16, and 20 are withdrawn in view of Applicant's amendments to the claims.
8. The 35 U.S.C. § 112, second paragraph, rejections of Claims 12-24 are withdrawn in view of Applicant's amendments to the claims.
9. The 35 U.S.C. § 101 rejections of Claims 1-8 and 17-24 are withdrawn in view of Applicant's amendments to the claims.

Response to Amendment

Claim Rejections - 35 USC § 112

10. The following is a quotation of the first paragraph of 35 U.S.C. 112:

The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same and shall set forth the best mode contemplated by the inventor of carrying out his invention.

11. **Claims 17-24** are rejected under 35 U.S.C. 112, first paragraph, as failing to comply with the written description requirement. The claim(s) contains subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

Claims 17-24 recite computer readable medium as a claimed element. The amendment made to the specification introduces new matter into the disclosure. 35 U.S.C. 132(a) states that no amendment shall introduce new matter into the disclosure of the invention. The added material which is not supported by the original disclosure is as follows: the definition of computer readable medium will be changed in view of the amendment.

Claim Objections

12. **Claim 18** is objected to because of the following informalities:

- **Claim 18** contains a typographical error: The word “medium” is missing after “computer readable.” It is noted that the word “medium” is in the previous version of the claim.

Appropriate correction is required.

Claim Rejections - 35 USC § 102

13. The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

A person shall be entitled to a patent unless –

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

14. **Claims 1-4, 6, 8, 9, 11-17, 19, and 20-24** are rejected under 35 U.S.C. 102(e) as being anticipated by **Barritz et al.** (US 6,519,766).

As per **Claim 1**, **Barritz et al.** disclose:

- a display device (*see Column 4: 32-35, “Computer program profiler 102 is executed by one or more processing computers. The processing computers can be mainframe computers, minicomputers or personal computers.”*);

Art Unit: 2191

- at least one processor (*see Column 4: 32-35, "Computer program profiler 102 is executed by one or more processing computers. The processing computers can be mainframe computers, minicomputers or personal computers."*);
- a computer readable storage medium comprising instructions executable by the at least one processor (*see Column 4: 32-35, "Computer program profiler 102 is executed by one or more processing computers. The processing computers can be mainframe computers, minicomputers or personal computers."*), instructions comprising instructions for:
 - a data engine adapted to identify profile data corresponding to low-level instances of a software application (*see Figure 1: 104; Column 4: 20-23, "... data gathering component 104 gathers data on the sequential execution logic and performance of a subject program 110."*);
 - a model library adapted to store at least one model, the at least one model having high-level instances (*see Column 4: 32-35, "Computer program profiler 102 is executed by one or more processing computers. The processing computers can be mainframe computers, minicomputers or personal computers."*; *Column 4: 24-27, "The data gathered on subject program 110 by data gathering component 104 is transferred to analysis component 106 for analysis and creation of a specialized data structure referred to as a "path map."*);
 - a model mapping engine adapted to at least one of query the data engine to obtain a list of the high-level instances, query the profile data, and map the profile data to the high-level instances (*see Figure 1: 106; Column 9: 51-54, "During the data reduction phase 502, the transaction and event data to be analyzed are selected and a data structure representing an internal map of the recorded events is created which relates the recorded events to one another."*); and

Art Unit: 2191

- a visualization system adapted to present the profile data in terms of the high-level instances on the display device (*see Figure 1: 108; Column 6: 23-25, "The output, displayed on a screen, printed, or stored for future use is a path diagram that reflects the actual processing flow of subject program 210."*; *Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602."*).

As per **Claim 2**, the rejection of **Claim 1** is incorporated; and Barritz et al. further disclose:

- wherein the visualization system is at least one of a sampling-based profile visualization system and a call graph profile visualization system (*see Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602."*).

As per **Claim 3**, the rejection of **Claim 2** is incorporated; and Barritz et al. further disclose:

- wherein the profile data is sampling-based profile data and the sampling-based profile visualization system is adapted to present the sampling-based profile data via an architecture view (*see Column 6: 58-64, "The initial task 302 performed by data gathering component 104 gets parameters from the user specifying the data to be gathered. For example, the user may be*

Art Unit: 2191

asked to specify the names of the transactions to be monitored for profiling.”; Column 41: 38-40, “In addition to the path map, other data, both captured and calculated can be presented to the user in the form of output tables, charts, graphs, reports, etc.”).

As per **Claim 4**, the rejection of **Claim 2** is incorporated; and Barritz et al. further disclose:

- wherein the profile data is call graph profile data and the call graph profile visualization system is adapted to present the call graph profile data via a hierarchical view (*see Figure 2; Column 10: 15-25, “... a specialized data structure, referred to as a “path map,” is build. The path map comprises a number of path element control blocks (“PECBs”) which are created from the event data selected in the first data reduction phase task 602.” and “The path formed by these PECBs is a logical map showing how each recorded event relates, in order of execution, to each other related event.”).*

As per **Claim 6**, the rejection of **Claim 1** is incorporated; and Barritz et al. further disclose:

- a model library browser adapted to at least one of create, edit, automatically generate, and select the at least one model (*see Figure 1: 106; Column 10: 15-25, “... a specialized data structure, referred to as a “path map,” is build. The path map comprises a number of path element control blocks (“PECBs”) which are created from the event data selected in the first data reduction phase task 602.”).*

Art Unit: 2191

As per **Claim 8**, the rejection of **Claim 1** is incorporated; and Barritz et al. further disclose:

- wherein the model mapping engine is adapted to perform at least one of a top-level instance query, a high-level instance structure query, a high-level instance flattening query, and a profile data query (*see Column 5: 14-17, "... query a database ..."*).

As per **Claim 9**, Barritz et al. disclose:

- mapping profile data of a software application to low-level instances of the software application (*see Column 9: 51-54, "During the data reduction phase 502, the transaction and event data to be analyzed are selected and a data structure representing an internal map of the recorded events is created which relates the recorded events to one another."*);

- performing at least one of generating and selecting at least one model appropriate for the software application, the at least one model having high-level abstractions (*see Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602."*);

- applying the at least one model to the profile data to map the low-level instances to the high-level abstractions (*see Column 10: 15-25, "Each PECB initially contains data for one recorded event, although the PECBs in the final path map are likely to contain data on more than one recorded event."*); and

- creating visualizations of the high-level abstractions (*see Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build."*).

As per **Claim 11**, the rejection of **Claim 9** is incorporated; and Barritz et al. further disclose:

- wherein said performing at least one of generating and selecting comprises at least one of creating a new model, editing an existing model, and automatically generating a model *(see Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602. ")*.

As per **Claim 12**, Barritz et al. disclose:

- collecting profile data of a software application *(see Column 4: 20-23, "... data gathering component 104 gathers data on the sequential execution logic and performance of a subject program 110. ")*;
- selecting at least one model to analyze the profile data, the at least one model having top-level instances *(see Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602. ")*;
- retrieving the top-level instances *(see Column 11: 58-60, "PECB chain building begins by obtaining an event record from the file of sorted and selected event records ... ")*;
- creating a root node for each top level instance *(see Column 11: 60-62, "... a PECB is built ... from the event record selected ... ")*;

Art Unit: 2191

- generating a hierarchical model for each root node, each hierarchical model having a plurality of child nodes (*see Column 11: 62-65, "The PECB is a multi-threadable control block which will contain data about collected occurrences of the subject event and will be linked to events which precede and/or follow it in the PECB chain."*);
- associating the profile data with the plurality of child nodes (*see Column 12: 39-43, "The event duration for each event is written into the PECB for that event. An "event count" is also written into each PECB indicating the number of times the associated event was executed (in the sequences represented by the PECB chain) in all the recorded instances of the transaction."*); and
- displaying each hierarchical model (*see Column 6: 23-25, "The output, displayed on a screen, printed, or stored for future use is a path diagram that reflects the actual processing flow of subject program 210."*).

As per **Claim 13**, the rejection of **Claim 12** is incorporated; and Barritz et al. further disclose:

- wherein the generating is done recursively (*see Column 12: 31-34, "Decision 910 creates a loop by which PECBs are built and chained as described above until a PECB for the last event for the current transaction instance has been built and chained."*).

As per **Claim 14**, the rejection of **Claim 12** is incorporated; and Barritz et al. further disclose:

Art Unit: 2191

- traversing each hierarchical model to obtain a list of functions within the software application (*see Column 12: 13-15, "The PECB will also contain a number of pointers for linking the PECB to other PECBs before and after it on the PECB chain."*); and

- creating a child node for each function (*see Column 12: 27-31, "Chaining is accomplished by writing a forward pointer in the pervious PECB which points to the current PECB, and by writing a backward pointer in the current PECB which points to the previous PECB in order to maintain the preferable two-way linkage."*).

As per **Claim 15**, the rejection of **Claim 12** is incorporated; and Barritz et al. further disclose:

- wherein the profile data is sampling-based profile data (*see Column 6: 58-64, "The initial task 302 performed by data gathering component 104 gets parameters from the user specifying the data to be gathered. For example, the user may be asked to specify the names of the transactions to be monitored for profiling."*).

As per **Claim 16**, the rejection of **Claim 12** is incorporated; and Barritz et al. further disclose:

- wherein the profile data is call graph profile data (*see Figure 2; Column 10: 15-25, "... a specialized data structure, referred to as a "path map," is build. The path map comprises a number of path element control blocks ("PECBs") which are created from the event data selected in the first data reduction phase task 602." and "The path formed by these PECBs is a*

logical map showing how each recorded event relates, in order of execution, to each other related event.”).

Claims 17 and 19 are computer readable medium claims corresponding to the method claims above (Claims 9 and 11) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 9 and 11.

Claims 20-24 are computer readable medium claims corresponding to the method claims above (Claims 12-16) and, therefore, are rejected for the same reasons set forth in the rejections of Claims 12-16.

Claim Rejections - 35 USC § 103

15. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

16. **Claims 5, 10, and 18** are rejected under 35 U.S.C. 103(a) as being unpatentable over **Barritz et al.** (US 6,519,766) in view of **Fagg, III et al.** (US 5,960,419).

As per **Claim 5**, the rejection of **Claim 1** is incorporated; however, **Barritz et al.** do not disclose:

- an expert system adapted to provide high-level advice relating to the low-level instances of the software application.

Fagg, III et al. disclose:

- an expert system adapted to provide high-level advice relating to the low-level instances of the software application (*see Column 1: 42-52, "Computer-based decision making systems, often known as expert systems, provide advice or opinion based on facts entered into the system by the user."*).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fagg, III et al. into the teaching of Barritz et al. to include an expert system adapted to provide high-level advice relating to the low-level instances of the software application. The modification would be obvious because one of ordinary skill in the art would be motivated to permit the user to exercise intellectual control over decisions made in the process of completing the task (*see Fagg, III et al. – Column 1: 20-22*).

As per **Claim 10**, the rejection of **Claim 9** is incorporated; however, Barritz et al. do not disclose:

- providing advice to improve performance of the software application in terms of the high-level abstractions.

Fagg, III et al. disclose:

- providing advice to improve performance of the software application in terms of the high-level abstractions (*see Column 1: 42-52, "Computer-based decision making systems, often*

Art Unit: 2191

known as expert systems, provide advice or opinion based on facts entered into the system by the user. ”).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Fagg, III et al. into the teaching of Barritz et al. to include providing advice to improve performance of the software application in terms of the high-level abstractions. The modification would be obvious because one of ordinary skill in the art would be motivated to permit the user to exercise intellectual control over decisions made in the process of completing the task (*see Fagg, III et al. – Column 1: 20-22*).

Claim 18 is rejected for the same reason set forth in the rejection of Claim 10.

17. **Claim 7** is rejected under 35 U.S.C. 103(a) as being unpatentable over Barritz et al. (US 6,519,766) in view of Ben-Romdhane et al. (US 2004/0031015).

As per **Claim 7**, the rejection of **Claim 6** is incorporated; however, Barritz et al. do not disclose:

- wherein the model library browser includes at least one of a model editor adapted to edit the at least one model, and a model generator adapted to generate the at least one model.

Ben-Romdhane et al. disclose:

- wherein the model library browser includes at least one of a model editor adapted to edit the at least one model, and a model generator adapted to generate the at least one model (*see Figure 1: 2 and 5; Paragraph [0056], “Source code 1 is analyzed by model generator 2 to*

Art Unit: 2191

create information model 3.” and “Information model 3 can also be enriched by model editor 5, which allows supporting information and documentation to be associated with information model 3.”).

Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to incorporate the teaching of Ben-Romdhane et al. into the teaching of Barritz et al. to include wherein the model library browser includes at least one of a model editor adapted to edit the at least one model, and a model generator adapted to generate the at least one model. The modification would be obvious because one of ordinary skill in the art would be motivated to divine any meaningful macro structure from the body of software (*see Ben-Romdhane et al. – Paragraph [0011]*).

Response to Arguments

18. Applicant’s arguments filed on June 20, 2007 have been fully considered, but they are not persuasive.

In the remarks, Applicant argues that:

a) Barritz does not disclose or suggest identifying profile data corresponding to low-level instances of a software application. In Barritz, the performance evaluation of a computer program is performed by gathering data on events, (Barritz, col. 4 lines 66- 67, col. 5 lines 1-19). Events may be "low-level" or "high-level" events (Barritz, col. 5, lines 1-19). Examples of events provided in Barritz include read file, write file, and network I/O. Barritz appears to disclose compiling performance data based on the occurrence of various defined events during the

Art Unit: 2191

runtime of a program. The data gathered corresponds to the sequential execution logic of the program, i.e., the instructions executed by the processor in order to run the program.

Events as disclosed in Barritz are wholly different from instances. An instance may be a structural element of a computer program. For example, a low-level instance may refer to an executable file, library file, or a function within such files. An event in Barritz involves the invocation of a particular command, such as, for example, the aforementioned write file. Barritz may track each occurrence of the invocation of the write file command as an event, and thus, may only have data for the write file command if an event involving the write file command occurs. If no such command occurs, there will be no event data involving the write file command in the data produced by Barritz. Unlike an event, a low-level instance is a structure that may be tracked independently of an event. If a program contains a specific low-level instance, for example, writefile.exe, and during the course of running the program writefile.exe is called 0 times, data will still be gathered on writefile.exe because it is an instance, not an event. Events in Barritz may only be used to gather data on what happens during the running of a program. Instances may be used to gather data on both what does and does not happen to each instance during the running of a program. Therefore, events as described in Barritz are wholly different from instances, and Barritz does not disclose or suggest "a data engine adapted to identify profile data corresponding to low-level instances of a software application."

Examiner's response:

- a) Examiner disagrees with Applicant's assertion that events are different from instances. The originally-filed specification does not provide an explicit and deliberate definition of what

Art Unit: 2191

an instance is. However, it does provide an exemplary definition of an instance as being processes and modules (*see Paragraph [0009]*). As noted by the Applicant, examples of low-level events provided in Barritz et al. include read file, write file, create database record, perform network I/O, and perform an operating system function. These examples of a low-level event clearly fit within the scope of the definition of an instance as defined by the originally-filed specification.

Furthermore, although the claims are interpreted in light of the specification, limitations from the specification are not read into the claims. See *In re Van Geuns*, 988 F.2d 1181, 26 USPQ2d 1057 (Fed. Cir. 1993).

Conclusion

19. **THIS ACTION IS MADE FINAL.** Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire THREE MONTHS from the mailing date of this action. In the event a first reply is filed within TWO MONTHS of the mailing date of this final action and the advisory action is not mailed until after the end of the THREE-MONTH shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the mailing date of this final action.

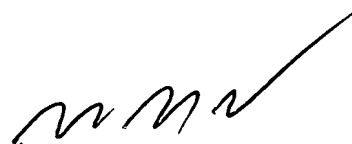
Art Unit: 2191

Any inquiry concerning this communication or earlier communications from the Examiner should be directed to Qing Chen whose telephone number is 571-270-1071. The Examiner can normally be reached on Monday through Thursday from 7:30 AM to 4:00 PM. The Examiner can also be reached on alternate Fridays.

If attempts to reach the Examiner by telephone are unsuccessful, the Examiner's supervisor, Wei Zhen, can be reached on 571-272-3708. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Any inquiry of a general nature or relating to the status of this application or proceeding should be directed to the TC 2100 Group receptionist whose telephone number is 571-272-2100.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).



WEI ZHEN
SUPERVISORY PATENT EXAMINER

QC / ac
July 13, 2007